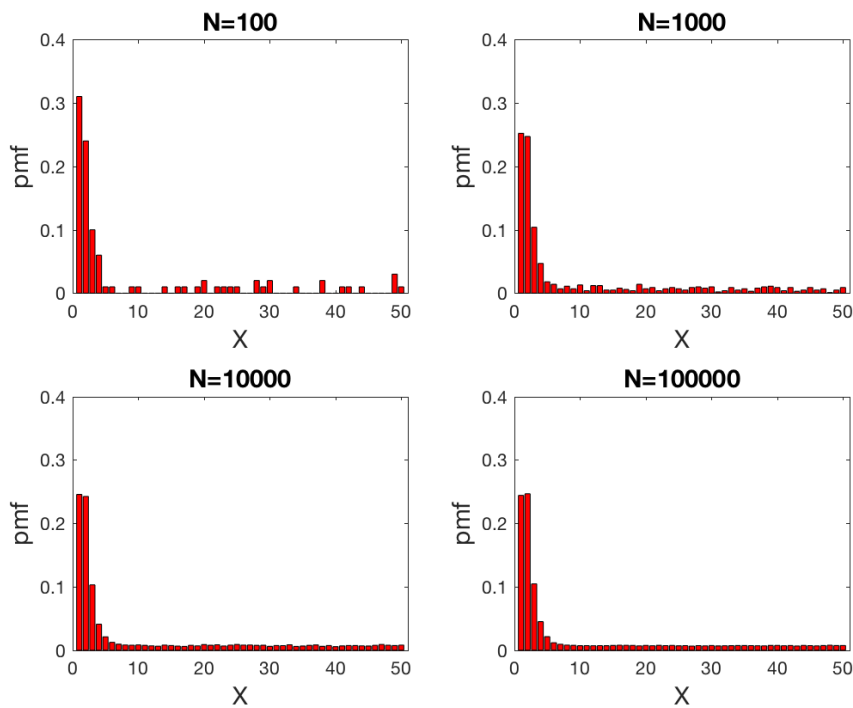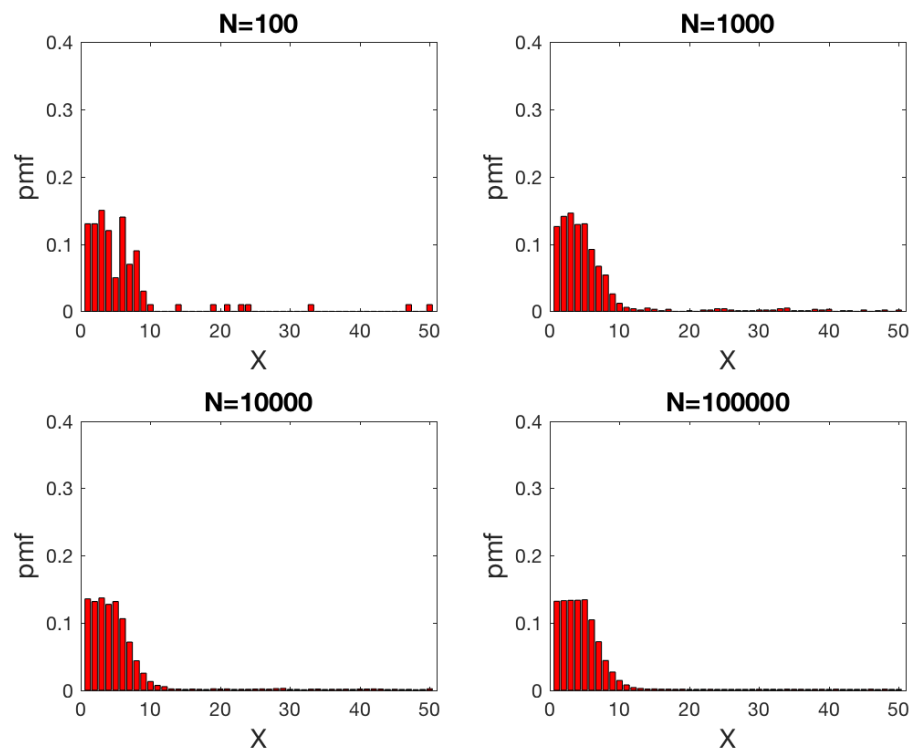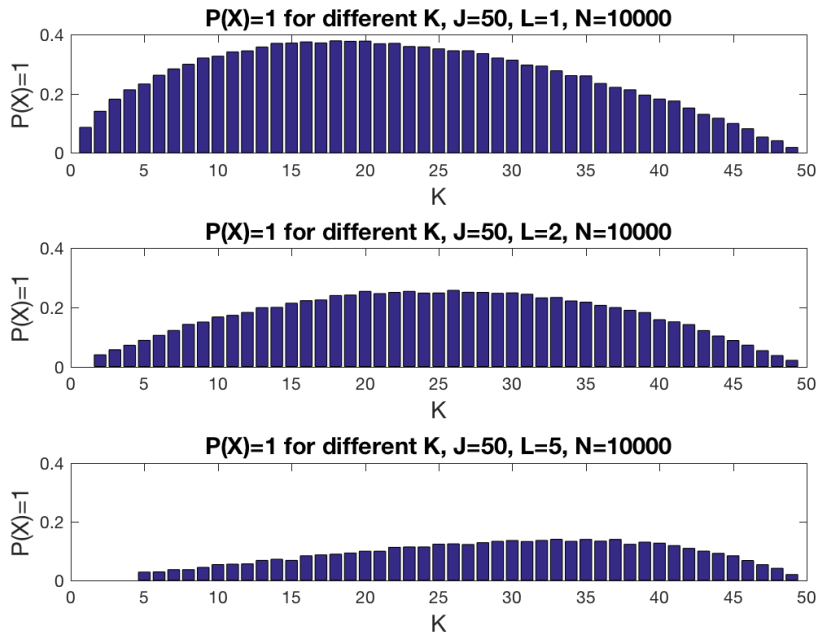This is the graph when setting L=2, we see that when setting L==2, the probability of accepting the first 3 bids are higher than 0.5 in total, which is a good signal.

The graph of L==5 suggests that when L gets larger the probability of selecting top L elements becomes approximately equal. Still the probability of selecting the top elements are high in total.

C)

P(X)=1 for different K, J=50, L=1, N=10000

P(X)=1 for different K, J=50, L=2, N=10000

P(X)=1 for different K, J=50, L=5, N=10000

Setting N==10000, the plot shows that as L grows larger, K should also be larger in order to capture the best bid. When L==1, the distribution is right skewed. When L==2, the distribution is almost symmetric, and when L==5 the distribution is right skewed.

CODE

```
function hw3
% questionA
% function defined as below as questionA

% questionB
J=50; K=30; L=2;
i=1;
for N=10.^[2 3 4 5]
    figure(1)
    subplot(2,2,i)
    questionB(J,K,L,N)
    i=i+1;
end
saveas(figure(1),strcat(pwd,'/figure1.png'))

close all
J=50; K=30; L=5;
i=1;
for N=10.^[2 3 4 5]
    figure(2)
    subplot(2,2,i)
    questionB(J,K,L,N)
    i=i+1;
end
saveas(figure(2),strcat(pwd,'/figure2.png'))
% questionC
close all
J=50;
i=1;
```

```matlab
for L=[1 2 5]
    figure(3)
    subplot(3,1,i)
    questionC(J,L)
    i=i+1;
end
saveas(figure(3),strcat(pwd,'/figure3.png'))

close all
end
% questionA
function [acep_rank,t_of_acceptance]= questionA(J,K,L)
    ofr=randperm(J);
    rej_ofr=ofr(1:K);
    sorted_rej_ofr=sort(rej_ofr);
    Lth_best_amongst_rej=sorted_rej_ofr(L);
    for i=K+1:J
        if (ofr(i)<Lth_best_amongst_rej)
            acep_rank=ofr(i);
            t_of_acceptance=i;
            return;
        end
    end
    acep_rank=ofr(J);
    t_of_acceptance=J;
end
% questionB
function []=questionB(J,K,L,N)
    accepted_ranks=zeros(1,N); % initialization of the vector of the ranks of
                               % the selected offers of the experiments!
    for i=1:N
        [accepted_ranks(i),ignr]=questionA(J,K,L);
    end
    [frequencies, ignr]=hist(accepted_ranks,J);
    pmf_vector=frequencies/N;
    bar(1:J,pmf_vector,'r')
    xlabel('X','FontSize',14)
    ylabel('pmf','FontSize',14)
    title(['N=',num2str(N)],'FontSize',14,'FontWeight','b')
    axis([0,51,0,0.4])
end
% questionC
function []=questionC(J,L)
N=10000;
KV=L:J-1;
probOfRank=zeros(1,J-L);
K_index=1;
for K=KV
    accepted_ranks=zeros(1,N);
    for i=1:N
        [accepted_ranks(i),ignr]=questionA(J,K,L);
    end
    [frequencies, ignr]=hist(accepted_ranks,J);
    pmf_vector=frequencies/N;
    probOfRank(1,K_index)=pmf_vector(1,1);
    K_index=K_index+1;
end
bar(KV,probOfRank)
    xlabel('K','FontSize',14)
    ylabel('P(X)=1','FontSize',14)
    title(['P(X)=1 for different K, J=',num2str(J),...
        ', L=', num2str(L),', N=',num2str(N)],'FontSize',...
        14,'FontWeight','b')
    axis([0,J,0,0.4])
end
```